ESCAPE

European Science Cluster of Astronomy &
Particle physics ESFRI research Infrastructures

Why FAIR and how to evaluate FAIRness

**Susana Sánchez Expósito, L. Verdes-Montenegro, J. Garrido,
M. Parra, J.R. Rodón, J. Pascual, R. Garrido**

Instituto de Astrofísica de Andalucía (CSIC)

EXCELENCIA SEVERO OCHOA

INSTITUTO de ASTROFISICA de ANDALUCIA CSIC I·A·A

CSIC CONSEJO SUPERIOR DE INVESTIGACIONES CIENTÍFICAS

WP5 Training Workshop 21-22 November 2022 (on-line)

ESCAPE - The European Science Cluster of Astronomy & Particle Physics ESFRI Research Infrastructures has received funding from the European Union's Horizon 2020 research and innovation programme under the Grant Agreement n° 824064.

Outline

- FAIR and Scientific Reproducibility
    - Bottom-up and up-bottom approach
- How to evaluate FAIRness for Research software
    - Metrics from the RDA WG
    - Methods and tools to evaluate FAIRness for software in practice
        - Badges
        - Automatic tools
        - Check lists
- Conclusions

Funded by the European Union's
Horizon 2020 - Grant N° 824064

FAIR and Scientific Reproducibility

Reproducibility is a fundamental principle of the Scientific Method … and not easy to achieve

Questionnaire on reproducibility (1500 scientists)
Baker (2016) https://doi.org/10.1038/533452a

- 70% of researchers have tried and failed to reproduce another scientist's experiments
- > 50% have failed to reproduce their own ones!
  - Chemistry: 90% (60%)
  - Biology: 80% (60%)
  - Physics and engineering: 70% (50%)
  - Medicine: 70% (60%)
  - Earth and environmental science: 60% (40%)

nature International weekly journal of science

Home | News & Comment | Research | Careers & Jobs | Current Issue | Archive | Audio & Video

Archive > Specials and supplements archive > Challenges in irreproducible research

SPECIAL

See all spe

CHALLENGES IN IRREPRODUCIBLE RESEARCH

Science moves forward by corroboration – when researchers verify others' results. Science

Funded by the European Union's
Horizon 2020 - Grant N° 824064

FAIR and Scientific Reproducibility

Some of the barriers:

- *Original data sets are not publicly available*
- *They are available but not in an automatic way*
- *Processed data is only available in the published PDF*
- *There are some scripts for processing the data on a server somewhere, but no one remembers where*
- *Code is in a public repository, but good luck trying to install/execute it*.

Bottom-up: Scientists see FAIR principles as a way to overcome those problems

Funded by the European Union's
Horizon 2020 - Grant N° 824064

Up - bottom

Beyond the mandate of publishing in Open Access

- EOSC is an initiative pursued by the EC since 2015

- Towards a reform of the research assessment system.
  (2021, https://data.europa.eu/doi/10.2777/707440)

  - *“Openness of research, and results that are* ***verifiable and reproducible*** *where applicable, strongly contribute to quality.”*

European Commission

European Research Area Policy Agenda

Overview of actions for the period 2022-2024

Funded by the European Union’s Horizon 2020 - Grant N° 824064

How to evaluate FAIRness: Metrics

Metrics for FAIR data

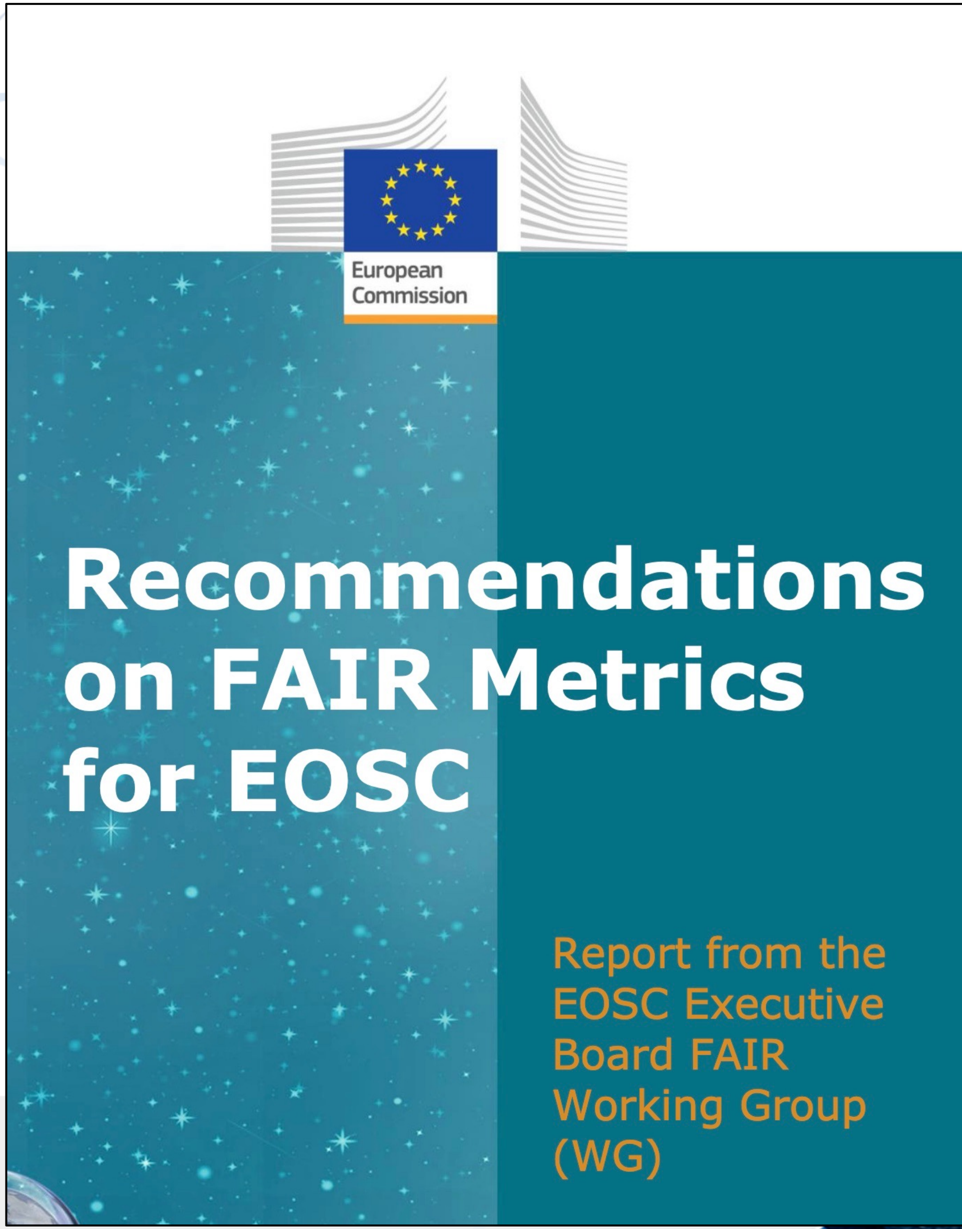
- FAIR Data Maturity Model Working Group (2020): FAIR Data Maturity Model. Specification and Guidelines. 10.15497/rda00050

- Metrics developed by FAIRsFAIR (10.5281/zenodo.3678715)

Metrics for FAIR research software

- Towards FAIR principles for research software (Lamprecht et al., 2019)

- FAIR Principles for Research Software (FAIR4RS Principles) 10.15497/RDA00068



European Commission

Recommendations on FAIR Metrics for EOSC

Report from the EOSC Executive Board FAIR Working Group (WG)

FAIR Principles for Research Software (FAIR4RS Principles)

10.15497/RDA00068

PIDs

Metadata

Access Protocols

F	The software should be easy to find for both humans and machines.
F1	Software is assigned a globally unique and persistent identifier.
F1.1	Different components of the software must be assigned distinct identifiers representing different levels of granularity.
F1.2	Different versions of the same software must be assigned distinct identifiers
F2	Software is described with rich metadata
F3	Metadata clearly and explicitly include the identifier of the software they describe
F4	Metadata are FAIR and are searchable and indexable
A	The software, and its metadata, must be retrievable via standardized protocols
A1	Software is retrievable by its identifier using a standardized communications protocol
A1.1	The protocol is open, free, and universally implementable.
A1.2	The protocol allows for an A&A procedure, where necessary
A2	Metadata are accessible, even when the software is no longer available.

Funded by the European Union's
Horizon 2020 - Grant N° 824064

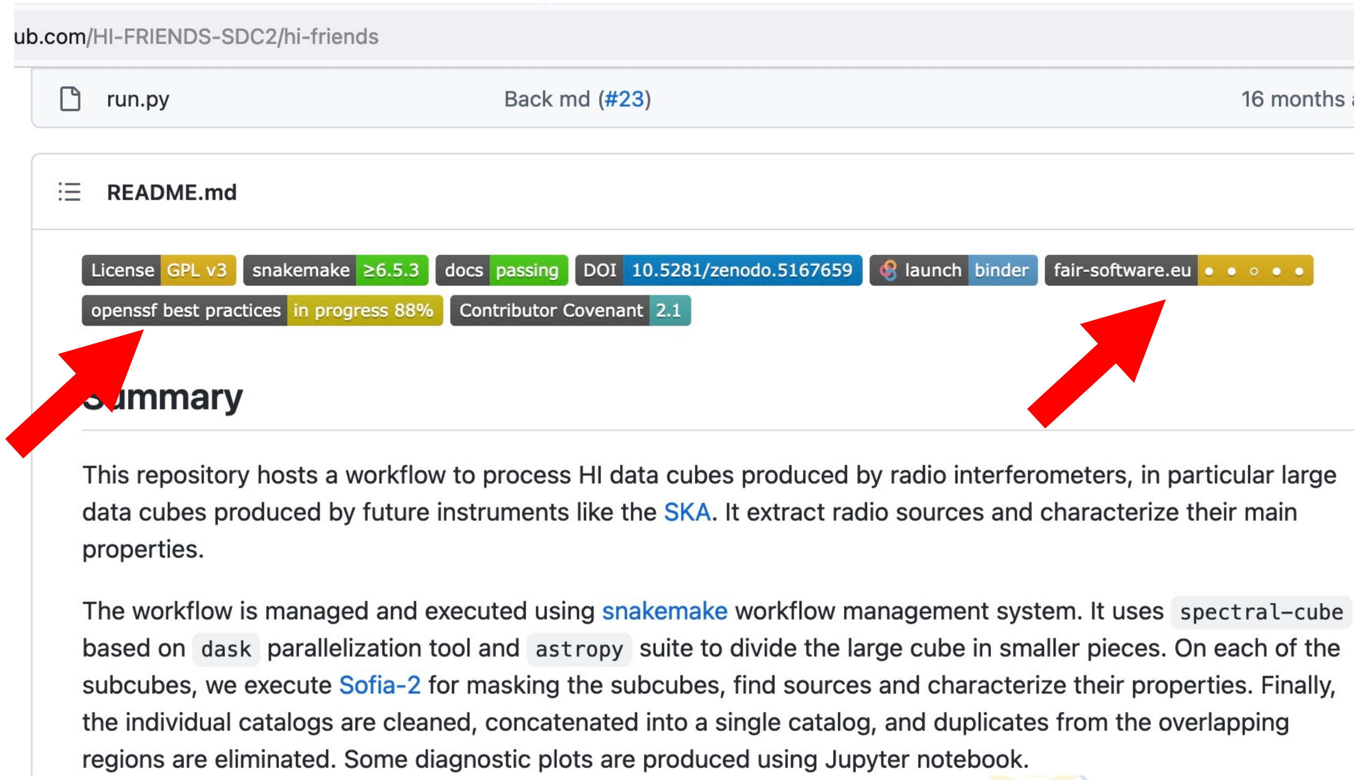
FAIR Principles for Research Software (FAIR4RS Principles)

10.15497/RDA00068

I	The software interoperates with other software through exchanging data and/or metadata, and/or through interaction via APIs.
I1	Software reads, writes and exchanges data in a way that meets domain-relevant community standards.
I2	Software includes qualified references to other objects (e.g. parameters file)
R	The software is both usable (it can be executed) and reusable (it can be understood, modified, built upon, or incorporated into other software)
R1	Software is described with a plurality of accurate and relevant attributes.
R1.1	Software must have a clear and accessible license.
R1.2	Software is associated with detailed provenance.
R2	Software includes qualified references to other software (e.g. dependencies).
R3	Software meets domain-relevant community standards.

Funded by the European Union's
Horizon 2020 - Grant N° 824064

FAIRness evaluation in practice



ub.com/HI-FRIENDS-SDC2/hi-friends

run.py Back md (#23) 16 months

README.md

License GPL v3 | snakemake ≥6.5.3 | docs passing | DOI 10.5281/zenodo.5167659 | launch binder | fair-software.eu ● ● ○ ● ●

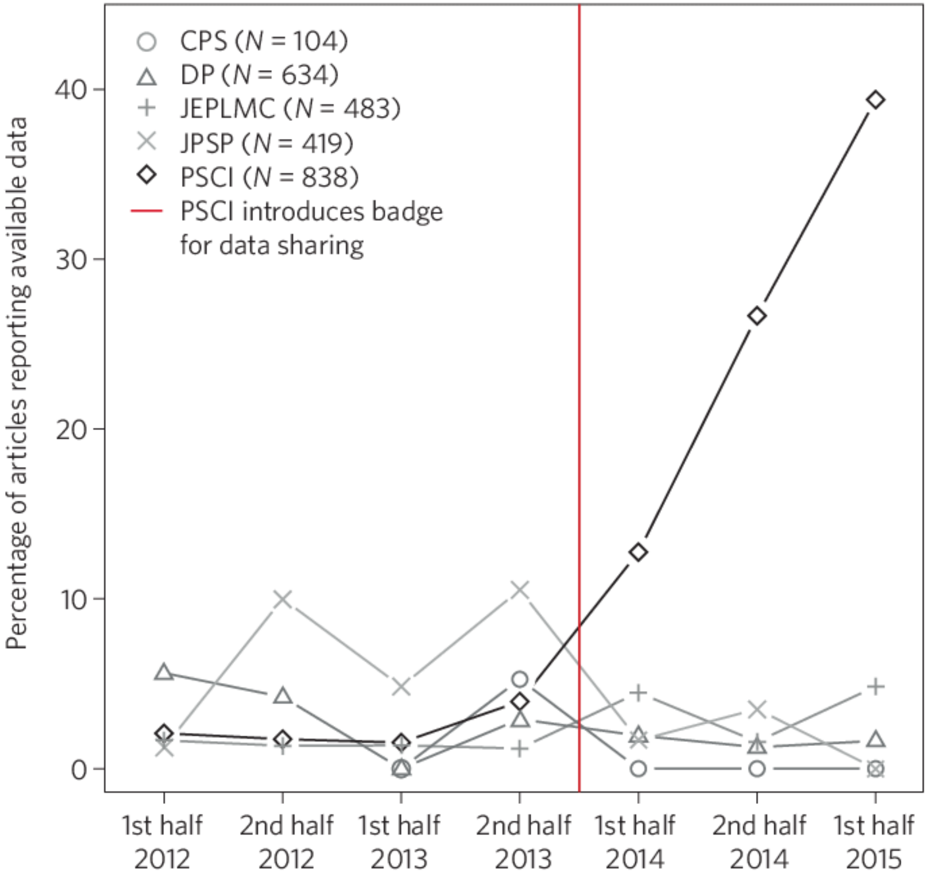
openssf best practices in progress 88% | Contributor Covenant 2.1

Summary

This repository hosts a workflow to process HI data cubes produced by radio interferometers, in particular large data cubes produced by future instruments like the SKA. It extract radio sources and characterize their main properties.

The workflow is managed and executed using snakemake workflow management system. It uses `spectral-cube` based on `dask` parallelization tool and `astropy` suite to divide the large cube in smaller pieces. On each of the subcubes, we execute Sofia-2 for masking the subcubes, find sources and characterize their properties. Finally, the individual catalogs are cleaned, concatenated into a single catalog, and duplicates from the overlapping regions are eliminated. Some diagnostic plots are produced using Jupyter notebook.

Badges as incentives



Source: M., Marcus et al. (2017). A manifesto for reproducible science. Nature Human Behaviour. 1. 0021. 10.1038/s41562-016-0021.

Funded by the European Union's Horizon 2020 - Grant N° 824064

FAIRness evaluation in practice

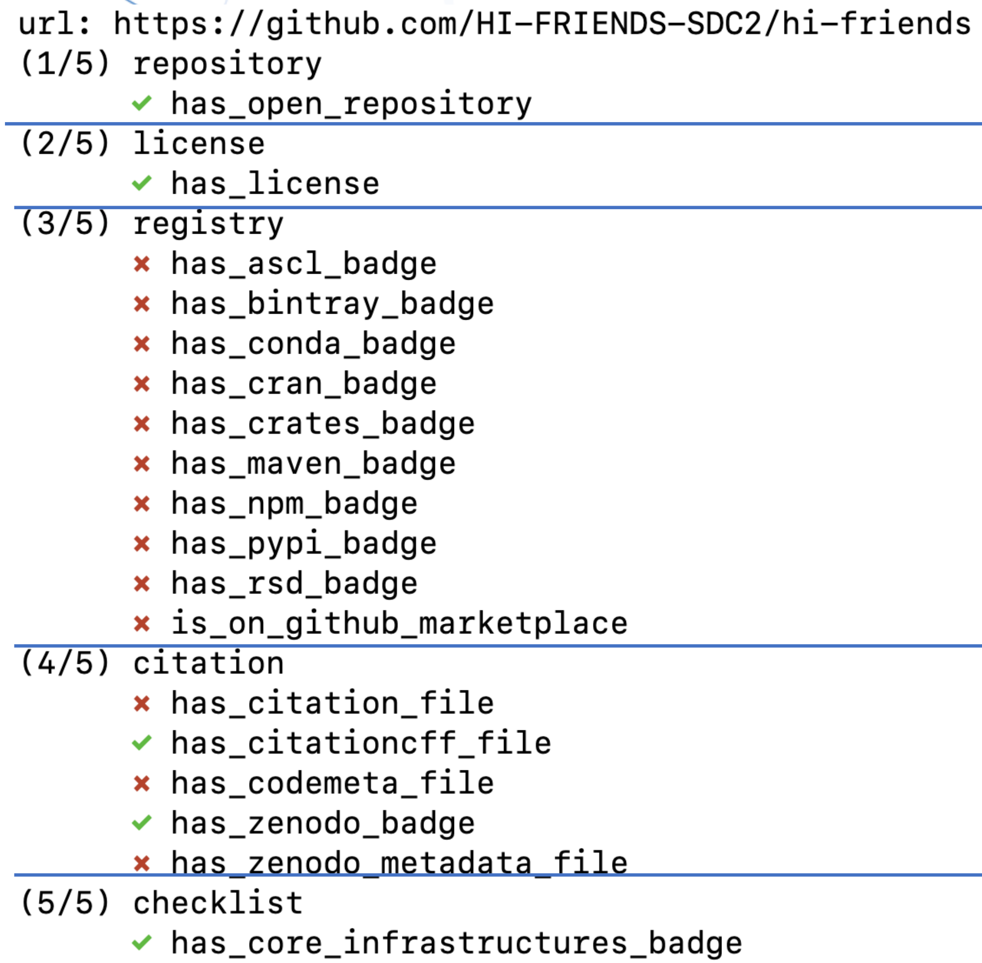
fair-software.eu ● ● ● ● ●

By the Netherlands eScience Center and the Dutch national centre of expertise and repository for research data

1. **Repository**: Is the software in a publicly accessible repository with version control?
2. **License**: Is there a license file? Use of standard licenses.
3. **Registry**: Is the software registered in one or more software registries?
4. **Citation**: Can the repository be cited easily? (CITATION.cff)
5. **Checklist**: Do the developers of the software use a software quality checklist? (e.g. OpenSSF Best Practices)

Source: https://github.com/fair-software/howfairis-github-action

(manual) ESAP evaluation according to fair-software.eu



```
url: https://github.com/HI-FRIENDS-SDC2/hi-friends
(1/5) repository
      ✓ has_open_repository
(2/5) license
      ✓ has_license
(3/5) registry
      × has_ascl_badge
      × has_bintray_badge
      × has_conda_badge
      × has_cran_badge
      × has_crates_badge
      × has_maven_badge
      × has_npm_badge
      × has_pypi_badge
      × has_rsd_badge
      × is_on_github_marketplace
(4/5) citation
      × has_citation_file
      ✓ has_citationcff_file
      × has_codemeta_file
      ✓ has_zenodo_badge
      × has_zenodo_metadata_file
(5/5) checklist
      ✓ has_core_infrastructures_badge
```

✓ Gitlab @ ASTRON repository

✓ Apache 2.0 for Gateway, GUI and Worker modules

✓ ESAP onboarded to the OSSR (zenodo)
fair-software.eu should include zenodo as a software registry

✓ Citation through DOI provided by Zenodo
Add a citation.cff file to the repository?

? Which best practices we should follow?

OpenSSF Best Practices Badge Program

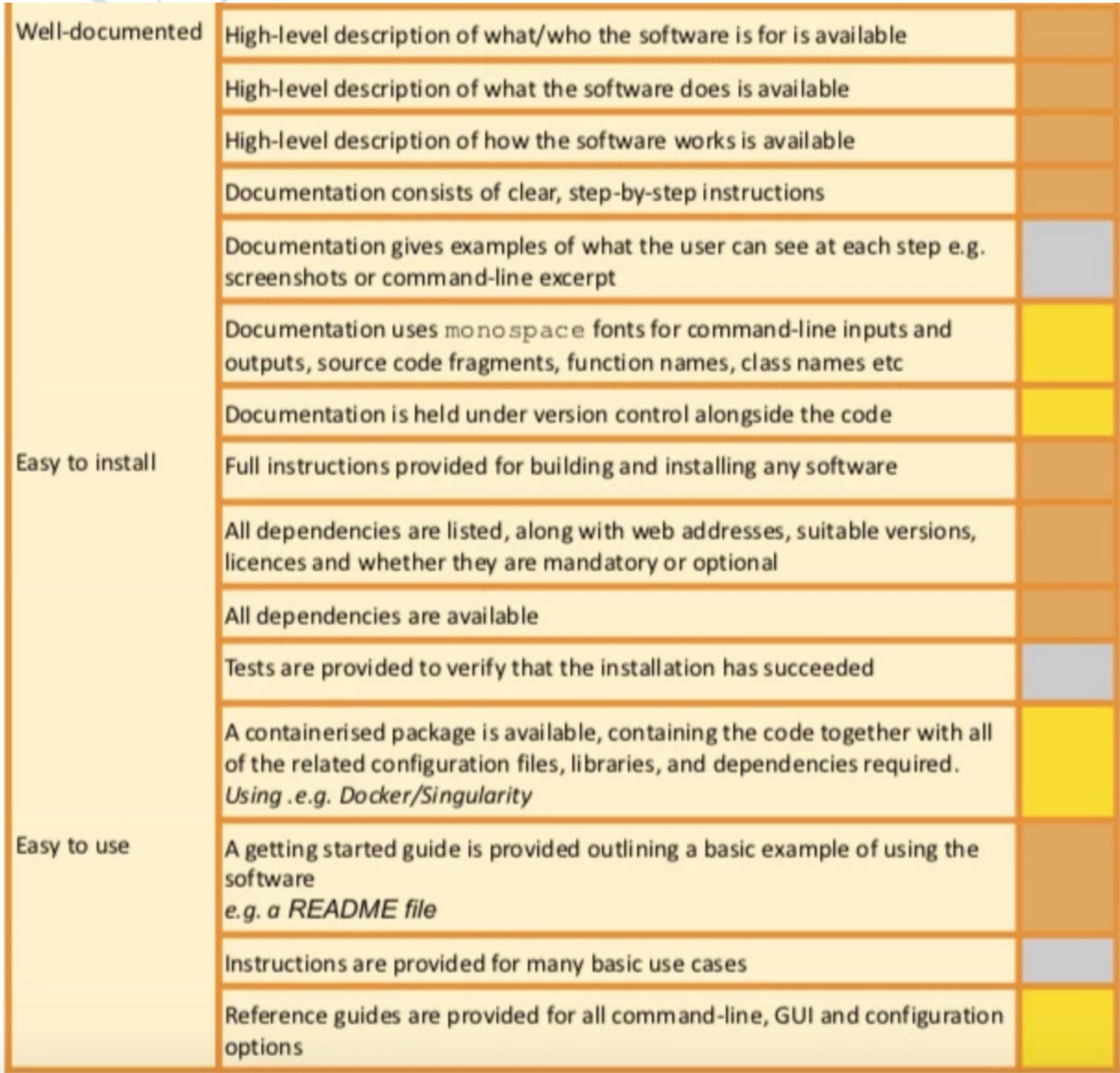
Funded by the European Union's
Horizon 2020 - Grant N° 824064

Software Best Practices for Scientific Reproducibility

Criteria to evaluate FAIRness in SKA Data Challenge 2

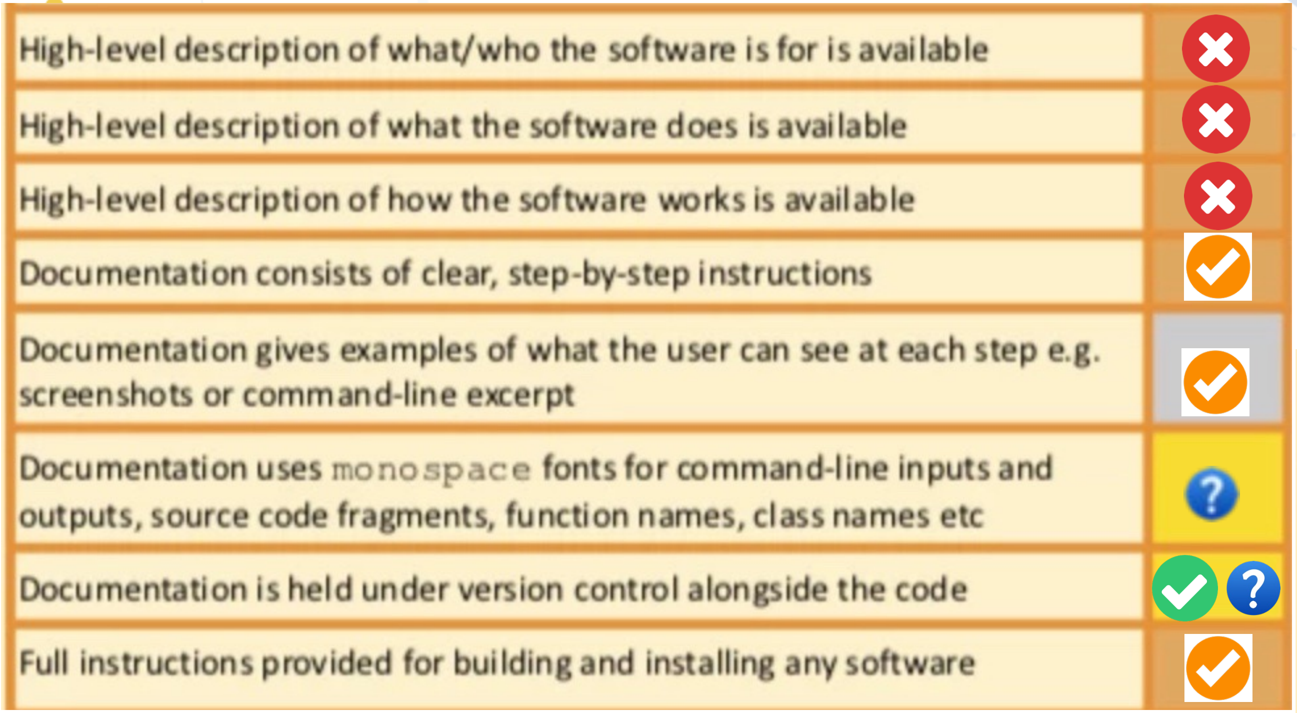
Can the pipeline be re-run easily to produce the same results?

1. **Well documented**
   - Who/What/how
   - Examples
   - Control version
2. **Easy to install**
   - Dependencies / Containers
   - Tests to verify the installation
3. **Easy to use**
   - Guides



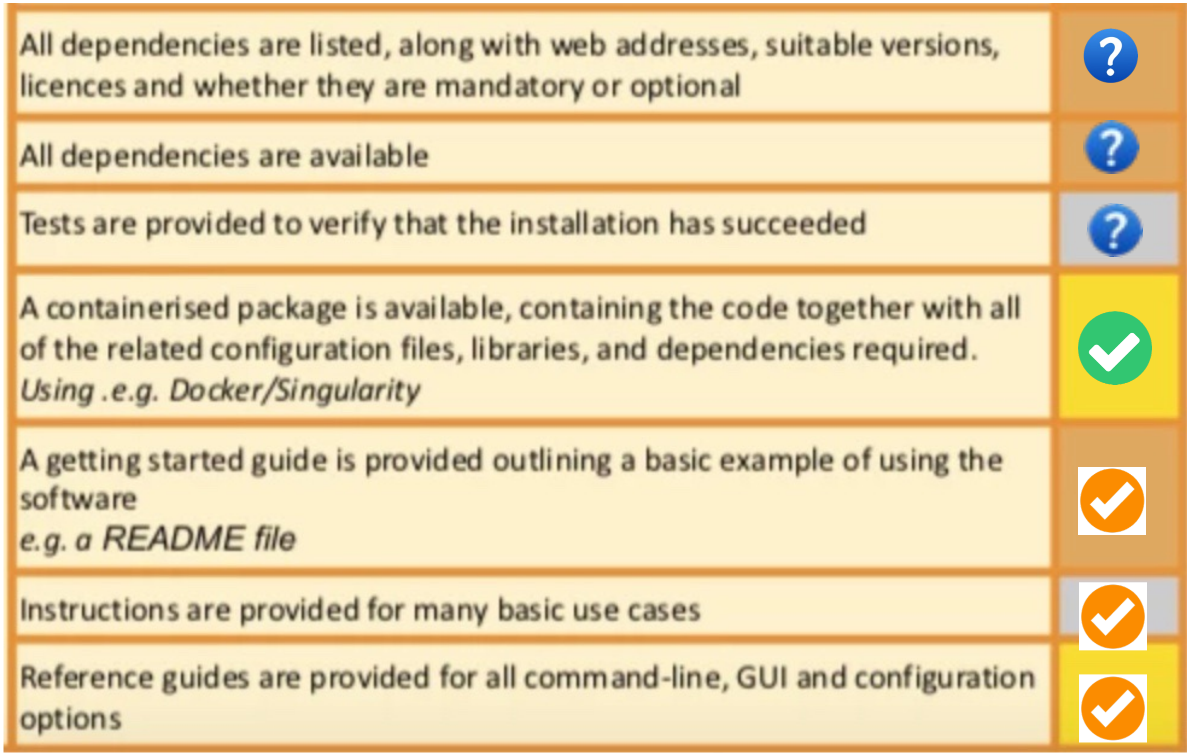
Well-documented	High-level description of what/who the software is for is available	🟧
	High-level description of what the software does is available	🟧
	High-level description of how the software works is available	🟧
	Documentation consists of clear, step-by-step instructions	🟧
	Documentation gives examples of what the user can see at each step e.g. screenshots or command-line excerpt	▨
	Documentation uses `monospace` fonts for command-line inputs and outputs, source code fragments, function names, class names etc	🟨
	Documentation is held under version control alongside the code	🟨
Easy to install	Full instructions provided for building and installing any software	🟧
	All dependencies are listed, along with web addresses, suitable versions, licences and whether they are mandatory or optional	🟧
	All dependencies are available	🟧
	Tests are provided to verify that the installation has succeeded	▨
	A containerised package is available, containing the code together with all of the related configuration files, libraries, and dependencies required. *Using .e.g. Docker/Singularity*	🟨
Easy to use	A getting started guide is provided outlining a basic example of using the software *e.g. a README file*	🟧
	Instructions are provided for many basic use cases	▨
	Reference guides are provided for all command-line, GUI and configuration options	🟨

Software Best Practices for Scientific Reproducibility



High-level description of what/who the software is for is available	❌
High-level description of what the software does is available	❌
High-level description of how the software works is available	❌
Documentation consists of clear, step-by-step instructions	✔
Documentation gives examples of what the user can see at each step e.g. screenshots or command-line excerpt	✔
Documentation uses `monospace` fonts for command-line inputs and outputs, source code fragments, function names, class names etc	❓
Documentation is held under version control alongside the code	✅ ❓
Full instructions provided for building and installing any software	✔

Documentation:
https://git.astron.nl/astron-sdc/escape-wp5/esap-api-gateway/-/wikis/home



All dependencies are listed, along with web addresses, suitable versions, licences and whether they are mandatory or optional	❓
All dependencies are available	❓
Tests are provided to verify that the installation has succeeded	❓
A containerised package is available, containing the code together with all of the related configuration files, libraries, and dependencies required. *Using .e.g. Docker/Singularity*	✅
A getting started guide is provided outlining a basic example of using the software *e.g. a README file*	✔
Instructions are provided for many basic use cases	✔
Reference guides are provided for all command-line, GUI and configuration options	✔

**** Preliminary ** evaluation for ESAP**

- ✅ Done
- ✔ Work in progress
- ❌ We did not identify the need yet
- ❓ Not sure if done nor if we want to do it

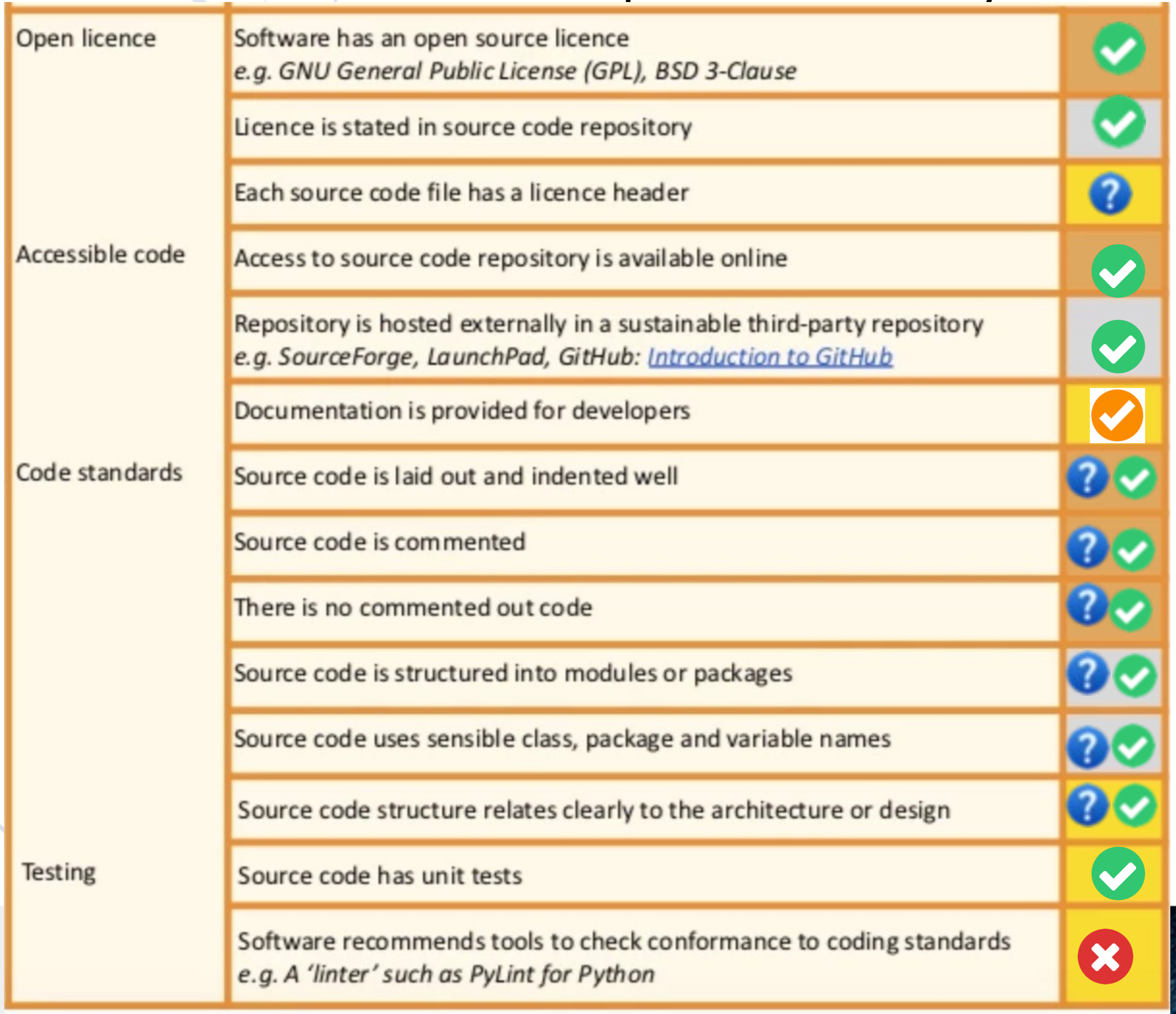
Funded by the European Union's
Horizon 2020 - Grant N° 824064

Software Best Practices for Scientific Reproducibility

Can the code be reused easily by other people to develop new projects?

1. **License**
   - Added in the code file header
2. **Accessible code**
   - Online repository
   - Documentation for developers
3. **Code standards**
4. **Testing**

Source: https://sdc2.astronomers.skatelescope.org/sdc2-challenge/reproducibility-awards

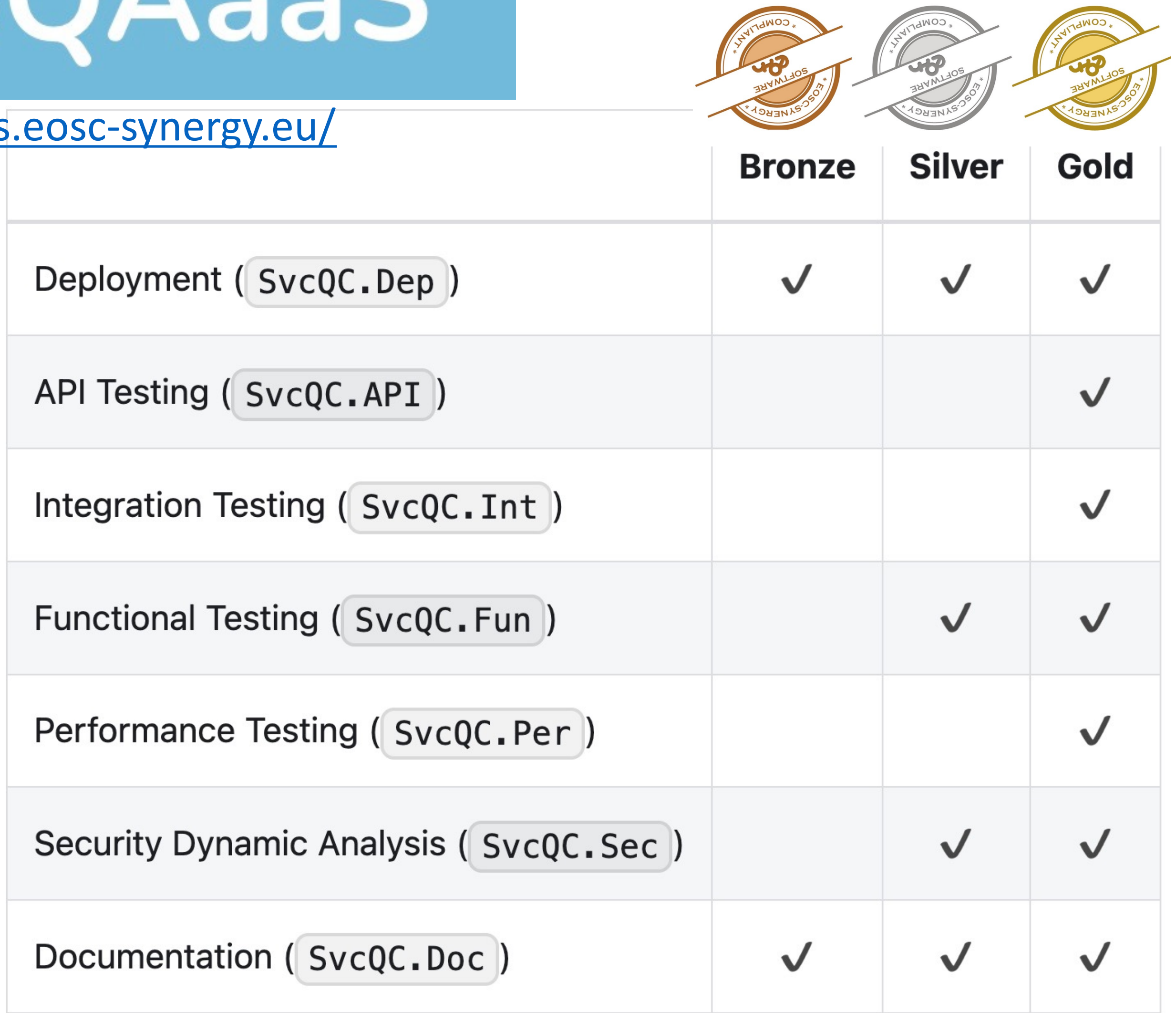


Open licence	Software has an open source licence *e.g. GNU General Public License (GPL), BSD 3-Clause*	✓
	Licence is stated in source code repository	✓
	Each source code file has a licence header	?
Accessible code	Access to source code repository is available online	✓
	Repository is hosted externally in a sustainable third-party repository *e.g. SourceForge, LaunchPad, GitHub: Introduction to GitHub*	✓
	Documentation is provided for developers	✓
Code standards	Source code is laid out and indented well	? ✓
	Source code is commented	? ✓
	There is no commented out code	? ✓
	Source code is structured into modules or packages	? ✓
	Source code uses sensible class, package and variable names	? ✓
	Source code structure relates clearly to the architecture or design	? ✓
Testing	Source code has unit tests	✓
	Software recommends tools to check conformance to coding standards *e.g. A 'linter' such as PyLint for Python*	✗

Source: https://sqaaas.eosc-synergy.eu/

Automatic evaluation using CI/CD pipelines

- A graphical tool to easily create CI/CD pipelines (user-customized evaluation)
- A tool to assess the quality of a software application



	Bronze	Silver	Gold
Deployment (`SvcQC.Dep`)	✓	✓	✓
API Testing (`SvcQC.API`)			✓
Integration Testing (`SvcQC.Int`)			✓
Functional Testing (`SvcQC.Fun`)		✓	✓
Performance Testing (`SvcQC.Per`)			✓
Security Dynamic Analysis (`SvcQC.Sec`)		✓	✓
Documentation (`SvcQC.Doc`)	✓	✓	✓

Criteria for Research Software
vs
Criteria for Services

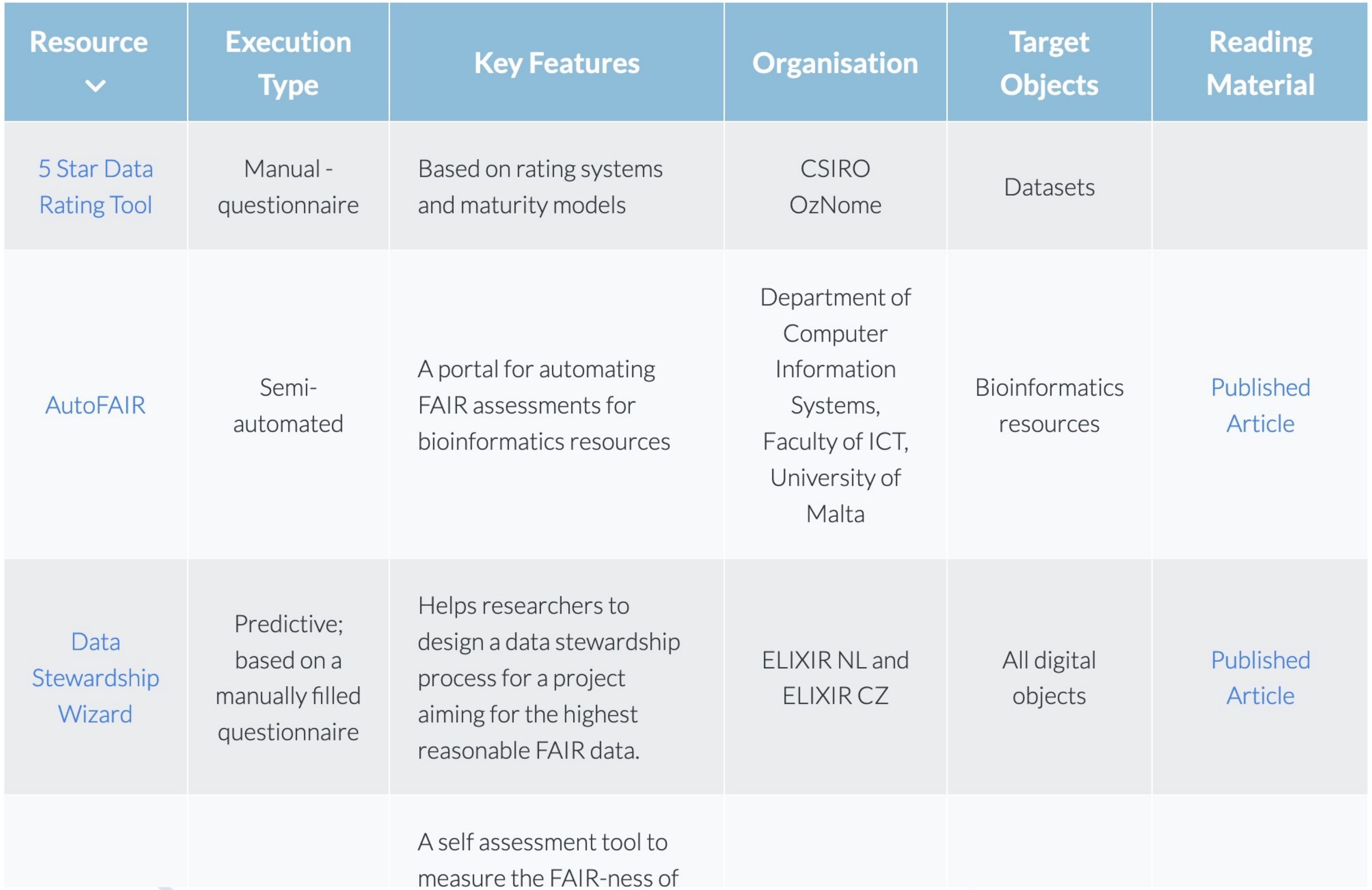
SQA Criteria for Services : https://doi.org/10.20350/digitalCSIC/12533

Funded by the European Union's Horizon 2020 - Grant N° 824064

Other tools for the assessment of digital objects against the FAIR principles

https://fairassist.org/

- Manual:
  - Questionnaire
  - Checklist
- Automated
- Research object
  - Data
  - Software
  - Others
- Badges



Resource	Execution Type	Key Features	Organisation	Target Objects	Reading Material
5 Star Data Rating Tool	Manual - questionnaire	Based on rating systems and maturity models	CSIRO OzNome	Datasets	
AutoFAIR	Semi-automated	A portal for automating FAIR assessments for bioinformatics resources	Department of Computer Information Systems, Faculty of ICT, University of Malta	Bioinformatics resources	Published Article
Data Stewardship Wizard	Predictive; based on a manually filled questionnaire	Helps researchers to design a data stewardship process for a project aiming for the highest reasonable FAIR data.	ELIXIR NL and ELIXIR CZ	All digital objects	Published Article
		A self assessment tool to measure the FAIR-ness of			

Funded by the European Union's
Horizon 2020 - Grant N° 824064

Conclusions

- FAIR metrics identify/ complement Quality Software Best Practices
- Up-bottom implementation (Being part of EOSC is also a kind of distinctive badge)
  - Recommendations on FAIR Metrics for EOSC
  - EOSC Task Forces output (Rules of Participation and Monitoring, FAIR metrics and Data quality)
  - EOSC Portal Onboarding Team output

- How to evaluate ESAP?
  - ESAP is not a Research Software
  - ESAP is a service (*a toolkit rather a running service*)
  - ESAP a service to support scientists in following Open Science practice
    - → How to evaluate this?

Funded by the European Union's
Horizon 2020 - Grant N° 824064

Thanks!

Funded by the European Union's
Horizon 2020 - Grant N° 824064

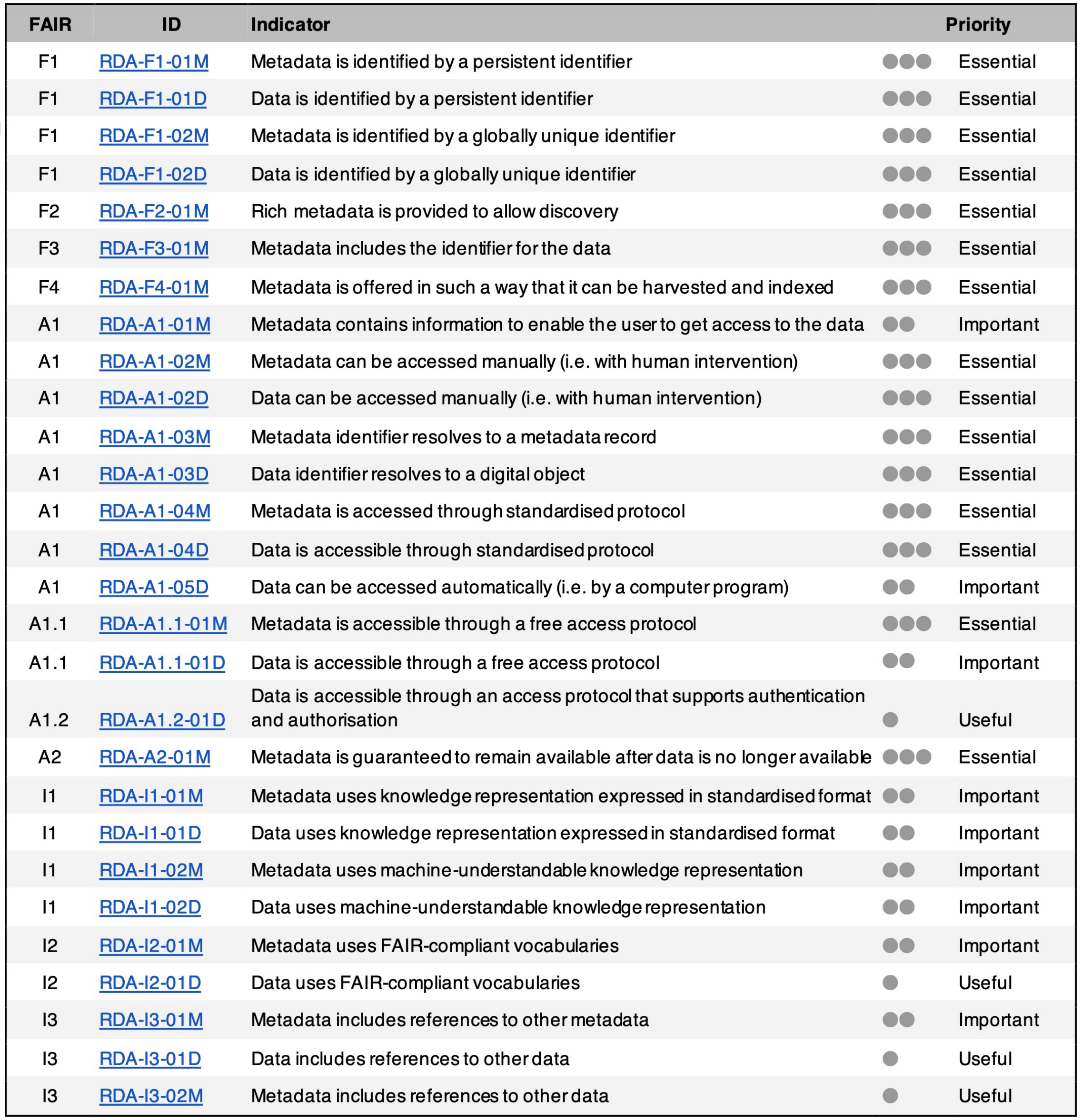
Extra slides

Funded by the European Union's
Horizon 2020 - Grant N° 824064

Metrics for FAIR data

Metrics developed by FAIRsFAIR (10.5281/zenodo.3678715)

- Universally Unique Identifier
- Persistent Identifier
- Descriptive Metadata
- Inclusion of Data Identifier in Metadata
- Searchable Metadata
- Data Access Level
- Metadata Preservation
- Semantic Representation of Metadata
- Qualified References to Related Entities
- Community-Driven Metadata
- Data Content Description
- Data Usage Licence
- Standard File Format



FAIR	ID	Indicator		Priority
F1	RDA-F1-01M	Metadata is identified by a persistent identifier	●●●	Essential
F1	RDA-F1-01D	Data is identified by a persistent identifier	●●●	Essential
F1	RDA-F1-02M	Metadata is identified by a globally unique identifier	●●●	Essential
F1	RDA-F1-02D	Data is identified by a globally unique identifier	●●●	Essential
F2	RDA-F2-01M	Rich metadata is provided to allow discovery	●●●	Essential
F3	RDA-F3-01M	Metadata includes the identifier for the data	●●●	Essential
F4	RDA-F4-01M	Metadata is offered in such a way that it can be harvested and indexed	●●●	Essential
A1	RDA-A1-01M	Metadata contains information to enable the user to get access to the data	●●	Important
A1	RDA-A1-02M	Metadata can be accessed manually (i.e. with human intervention)	●●●	Essential
A1	RDA-A1-02D	Data can be accessed manually (i.e. with human intervention)	●●●	Essential
A1	RDA-A1-03M	Metadata identifier resolves to a metadata record	●●●	Essential
A1	RDA-A1-03D	Data identifier resolves to a digital object	●●●	Essential
A1	RDA-A1-04M	Metadata is accessed through standardised protocol	●●●	Essential
A1	RDA-A1-04D	Data is accessible through standardised protocol	●●●	Essential
A1	RDA-A1-05D	Data can be accessed automatically (i.e. by a computer program)	●●	Important
A1.1	RDA-A1.1-01M	Metadata is accessible through a free access protocol	●●●	Essential
A1.1	RDA-A1.1-01D	Data is accessible through a free access protocol	●●	Important
A1.2	RDA-A1.2-01D	Data is accessible through an access protocol that supports authentication and authorisation	●	Useful
A2	RDA-A2-01M	Metadata is guaranteed to remain available after data is no longer available	●●●	Essential
I1	RDA-I1-01M	Metadata uses knowledge representation expressed in standardised format	●●	Important
I1	RDA-I1-01D	Data uses knowledge representation expressed in standardised format	●●	Important
I1	RDA-I1-02M	Metadata uses machine-understandable knowledge representation	●●	Important
I1	RDA-I1-02D	Data uses machine-understandable knowledge representation	●●	Important
I2	RDA-I2-01M	Metadata uses FAIR-compliant vocabularies	●●	Important
I2	RDA-I2-01D	Data uses FAIR-compliant vocabularies	●	Useful
I3	RDA-I3-01M	Metadata includes references to other metadata	●●	Important
I3	RDA-I3-01D	Data includes references to other data	●	Useful
I3	RDA-I3-02M	Metadata includes references to other data	●	Useful

Credits: FAIR Data Maturity Model Working Group (2020): FAIR Data Maturity Model. Specification and Guidelines. DOI: 10.15497/rda00050

Funded by the European Union's
Horizon 2020 - Grant N° 824064